Алгоритмы слепого разделения источников в пакетном режиме[$]

Д. В. Савостьянов[©]

При анализе числовых данных, изменяющихся во времени, часто есть основания считать, что наблюдаемые последовательности с достаточной точностью могут быть выражены линейными комбинации нескольких независимых компонент. Задача отыскания независимых компонент по заданной смеси называется слепым разделением сигналов. Для ее решения разработано множество методов, которые можно отнести к одному из двух классов — методы, использующие в процессе расчета весь массив данных целиком, и методы, основанные на диагонализации статистик, полученных по исходным данным. В этой статье мы показываем, что только методы второго типа допускают «пакетный» режим обработки данных, то есть могут эффективно применяться в случае, когда сигналы не могут храниться в доступной оперативной памяти во время проведения расчета.

1. Введение

1.1. История задачи. Возможно, наиболее классическим примером задачи, для решения которой требуется разделить наблюдаемые числовые последовательности на набор независимых источни-

[$]Работа выполнена при поддержке грантов РФФИ 05-1-00721, 06-01-08052 и Программы приоритетных фундаментальных исследований Отделения математических наук РАН «Вычислительные и информационные проблемы решения больших задач» по проекту «Матричные методы и технологии для задач со свербольшим числом неизвестных».

[©]Институт вычислительной математики РАН

ков, является «задача о вечеринке». На шумной вечеринке, где одновременно разговаривает множество людей, человек может концентрировать внимание на беседе с товарищем и игнорировать речь всех остальных; однако услышав из дальнего конца зала свое имя, он распознает его и реагирует на призыв. Эта психофизическая способность была впервые описана в 1953 году Колином Черри [1], изучавшим проблемы, встающие перед диспетчерами крупных аэропортов, когда приходящие сообщения от нескольких пилотов смешиваются в динамике. В своей статье Черри заключил, что «способность разделять звуки и игнорировать посторонние шумы основана на характеристиках звука, таких как тон говорящего, направление, с которого приходит звук, высота звука, частота речи». Таким образом было сформулировано предположение о том, что мозг разделяет поступающую смесь звуковых сигналов, основываясь на различии в структуре исходных компонент. Механизм разделения сигналов в человеческом мозге до сих пор не изучен полностью, однако алгоритмы решения родственных задач с помощью компьютеров нашли широкое применение в области цифровой обработки сигналов, где и получили название методов (слепого) разделения сигналов. На их основе решается множество практических задач в самых различных областях:

- медицинское исследование функций мозга на основе данных магнитной энцефалографии, магнитно-резонансной томографии или спектроскопии (см., напр. [2, 3, 4, 5, 6]);
- развитие методов передачи информации через MIMO (multi-input multi-output) каналы (см., напр. [7, 8, 9]);
- анализ смесей в химии и спектрографии (см. обзор [10]);
- исследование числовых рядов в финансовой математике [11, 12], социологии, статистике, при анализе спроса;
- контроль транспортных и производственных процессов;
- мультимедийные приложения, распознавание текстов и изображений [13, 14].

Математическая постановка задачи во всех этих приложениях весьма схожа и довольно проста: предполагается, что наблюдаемые *источники* $y_i(t),\ i = 1,\dots m$, линейно выражаются через *независимые компоненты* $x_j(t),\ j = 1,\dots,n$. Не ограничивая общности, можно считать, что и те, и другие имеют нулевое среднее. В матрично-векторной записи

$$y(t) = Ax(t) + \xi(t), \qquad (*)$$

векторы $y(t)$ и $x(t)$ называются соответственно векторами сигналов и источников, матрица A называется *смешивающей* и предполагается не зависящей от параметра t (дискретного или непрерывного), называемого временем. В наиболее общем случае никакой априорной информации о матрице A и характеристиках источников $x_i(t)$ не имеется — тогда говорят о методах *слепого* разделения сигналов (blind source separation, BSS), которым и посвящена наша статья. Компоненты вектора *шума* $\xi_i(t)$ полагаются независимыми и, как правило, нормально распределенными. Источники $x_i(t)$ предполагаются *независимыми,* однако при решении задачи о разделении сигналов это условие может быть по-разному использовано и отображено на алгоритм.

1.2. Классификация существующих методов решения. В простейшем случае независимость сигналов формулируется как их некоррелированность, то есть диагональность матрицы $E[x(t)x^*(t)]$, где $E[\,\cdot\,]$ — осреднение по времени. Тогда можно записать (для простоты изложения полагаем, что шумов нет)

$$\Phi_y = E[y(t)y^*(t)] = E[Ax(t)x^*(t)A^*] = AE[x(t)x^*(t)]A^* = ADA^*,$$

учитывая, что A не зависит от времени. Таким образом, A можно определить через решение задачи диагонализации матрицы ковариации. Один ответ очевиден — построить собственное разложение $\Phi_y = U\Lambda U^*$ и использовать в качестве A матрицу собственных векторов, а в качестве D — собственные значения Λ. Это решение не единственно (в качестве A может быть взята матрица вида

$$A = U\Lambda^{1/2}W\Lambda^{\dagger/2}, \qquad (1)$$

где W произвольная унитарная). В самом деле, матрица U унитарна, а смешивающая матрица A — не обязательно. Впрочем, описанный нами сейчас метод *главных компонент* (principal component analysis, $\mathcal{PCA}$), несмотря на свою простоту, широко применяется для понижения размерности при статистической обработке больших массивов данных (см. напр., [15, 16, 17]) и по сей день используется для предобработки (так называемого «обеления») входных данных в большинстве методов слепого разделения сигналов.

Мы поняли, что одной некоррелированности источников для разделения источников недостаточно, и требуется использовать какие-то дополнительные средства описания их независимости. В зависимости от выбора этих средств, итоговый алгоритм можно отнести к одному из следующих семейств.

- Алгоритмы, в которых независимые компоненты выбираются в направлениях, максимизирующих степень их *негауссовости,* минимизирующих меру *взаимной информации* или же максимизирующих *критерий правдоподобия* в линейном пространстве данных, возможно с дополнительными ограничениями. К этому семейству относятся алгоритмы Fast$\mathcal{ICA}$ [18], $\mathcal{MILCA}$ [22], $\mathcal{SNICA}$. Обзоры можно найти в [20, 21].

- Алгоритмы, в которых дополнительная информация о независимости извлекается за счет одновременной диагонализации *нескольких* статистик и/или использовании кумулянтов высокого порядка [23, 25]. При этом перывми вычисляются не независимые компоненты, а смешивающая матрица. К этому семейству относится, например, метод JADE.

С вычислительной точки зрения различие состоит в следующем. Алгоритмы первой группы вычисляют непосредственно независимые компоненты, итерационно адаптируя *весовой вектор* w (или всю *размешивающую матрицу* W) так, чтобы функции $x_i(t) = (w_i, y(t))$ давали экстремум функционала, выбранного в качестве статистического описания независимости. Таким образом, в ходе итерационного процесса требуется хранить в памяти весь

вектор данных *целиком* и на каждом шаге вычислять на его основе градиент целевой функции. Это может быть затруднительно в том случае, если объем данных слишком велик, и они не могут одновременно присутствовать в оперативной памяти (например, если речь идет о вычислениях в реальном времени на небольшом вычислительном узле). В этом случае разделение сигналов должно происходит порциями, помещающимися в оперативную память, а уменьшение размера такой порции сказывается на статистических свойствах выборки и итоговой точности разделения.

Алгоритмы второй группы работают не с самими сигналами, а только лишь с их статистиками. Центральная составляющая этих методов — одновременная диагонализация нескольких матриц статистик, причем их число может быть и достаточно большим. Алгоритмы решения этой задачи предлагались в работах [24, 26, 27, 28]; кроме того, поскольку задача одновременной диагонализации матриц эквивалентна задаче о трилинейном разложении тензора (трехмерного массива), для решения может быть использован любой алгоритм вычисления трилинейной аппроксимации тензора, со многими из которых можно познакомиться в [29]–[36]. Однако популярность этой группы методов у инженеров-вычислителей, по всей видимости, ниже, чем методов типа Fast$\mathcal{ICA}$. Об этом говорит как сравнительно небольшое количество отчетов об успешном использовании методов одновременной диагонализации для решения задачи $\mathcal{BSS}$, так и отсутствие подобных алгоритмов в доступных пакетах библиотечных программ. Вероятно, это связано с определенными трудностями при переходе с теоретико–вероятностного или статистического языка описания на язык линейной алгебры и матричной и тензорной арифметики. Однако эти методы кажутся нам очень перспективными, в частности при разделении сигналов в «пакетном» режиме, то есть в ситуации, когда невозможно разместить весь имеющийся массив данных целиком в оперативной памяти.

В этой статье мы рассмотрим вопросы реализации алгоритмов слепого разделения сигналов в пакетном режиме. В целях связности изложения мы кратно изложим схемы типовых алгоритмов из первой и второй группы, но для упрощения изложения будем пре-

небрегать наличием шумов в модели $(*)$.

2. Метод FastICA

Алгоритм изложен по статье [19].

2.1. Проецирование на базис главных компонент. Как показано в первом разделе, требование некоррелированности источников позволяет определить часть информации о смешивающей матрице, записав ее в виде (1) и преобразовав модель $(*)$ к виду

$$U\Lambda^{1/2}W\Lambda^{\dagger/2}x(t) = y(t),$$

где Λ и U — собственные значения и собственные вектора матрицы корреляции Φ_y, $\Lambda^\dagger$ — псевдообратная матрица, W — неизвестная унитарная матрица. Обратив первые два сомножителя и переобозначив $x(t) := \Lambda^{\dagger/2}x(t)$ (источники определены с точностью до множителя и порядка), получаем линейную модель с *унитарной* смешивающей матрицей

$$Wx(t) = z(t), \qquad \text{где} \quad z(t) = \Lambda^{\dagger/2}U^*y(t). \tag{2}$$

В методе FastICA унитарность смешивающей матрицы строго необходима, так как она позволяет находить независимые компоненты по-одной. Предварительное «обеление» сигналов по формуле (2) часто применяется и в других алгоритмах BSS, так как оно увеличивает устойчивость решаемой задачи к шумам и ошибкам округления.

Отметим, что при вычислении псевдообратной матрицы можно считать нулевыми все диагональные элементы, меньшие некоторого порогового значения, сузив тем самым задачу с пространства, образованное линейной оболочкой сигналов, на подпространство, образованное линейной оболочкой старших компонент (сигнальное подпространство), и отфильтровав подпространство младших компонент (шумовое). Во многих приложениях, например, при анализе больших массивов схожих изображений (распознавание портретных фотографий, отпечатков пальцев) это сужение приводит к значительной экономии памяти, требуемой для хранения информации. В

финансовой математике при анализе динамики курсов акций метод $\mathcal{PCA}$ тоже позволяет сжать массив данных, но что еще важнее — определить количество независимых факторов, действующих на рынок. На рис. 1 показан пример задачи, в которой данные о курсах 33 акций были с относительной точностью около одного процента приближены линейной комбинацией трех главных компонент.

2.2. Критерий независимости и меры негауссовости. Некоррелированость источников теперь выполнена автоматически, и нам требуется некоторое дополнительное описание независимости для определения W. Отметим, что переход в базис главных компонент позволяет определять независимые компоненты по-отдельности.

$$x(t) = W^* z(t), \quad x_i(t) = \sum_{j=1}^{m} \bar{w}_{ji} z_j(t) = (w_i, z(t)), \quad i = 1, \ldots, n,$$

где весовой вектор w_i определяет i-й столбец матрицы W.

Полученная формула определяет независимую компоненту $x_i(t)$ как линейную комбинацию главных компонент, которые в свою очередь являются линейной комбинацией наблюдаемых сигналов. Будем рассматривать компоненты $x_i(t)$ как реализации во времени некоторых случайных величин, а $y_i(t)$ и $z_i(t)$ как линейную комбинацию (смесь) этих случайных величин. Классический результат теории вероятности — центральная предельная теорема — говорит, что функция распределения суммы двух случайных величин всегда ближе (строго говоря, не дальше) к нормальному распределению, чем функции распределения слагаемых. Отметим, что линейная комбинация главных компонент с произвольными весами w может быть представлена в виде

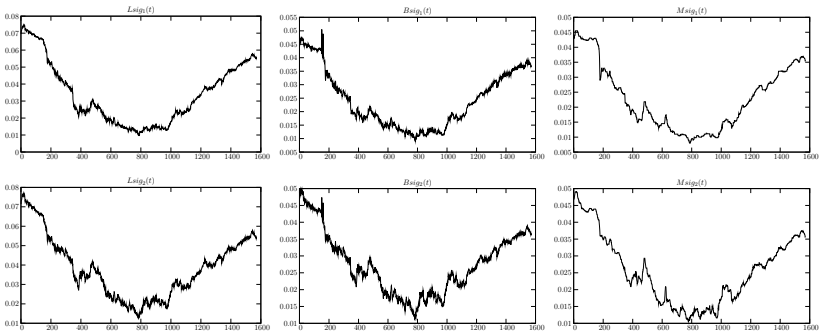
$$\xi = w^* z = w^* W x = q^* x, \qquad q = W^* w.$$

Пусть $\mathfrak{M}(\xi)$ — некоторая *мера негауссовости* случайной величины ξ (пока нам не важно, как именно выбирать эту меру). Максимум

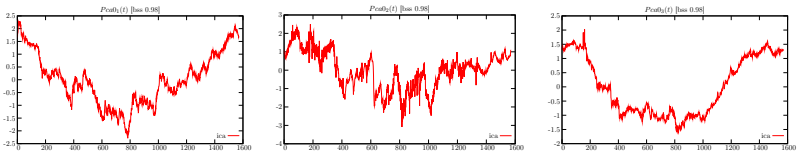
$$\mathfrak{M}(\xi) = \mathfrak{M}(q^* x) = \mathfrak{M}\left(\sum_{i=1}^{n} \bar{q}_i x_i\right)$$

Рис. 1. Применение метода главных компонент для понижения размерности при описании курсов акций

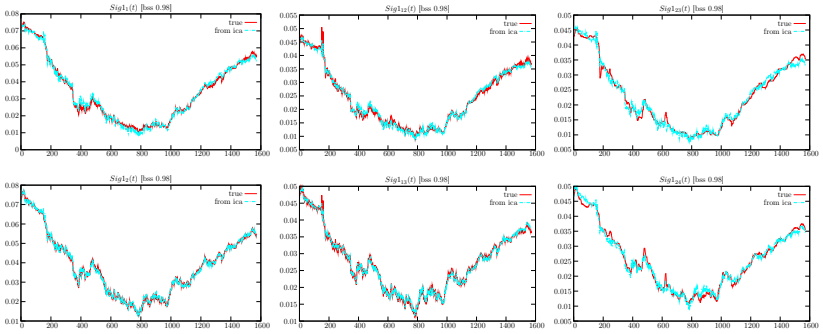
Курсы различных акций (всего 33, на рисунках 6)



Старшие главные компоненты (сигнальное пространство)



Проекции сигналов на сигнальное пространство



достигается, когда в правой части равенства стоит всего лишь одно слагаемое x_i. При этом весовой вектор как-то нормирован, например $\|w\| = 1$, а в силу унитарности матрицы W это означает, что и вектор q имеет нормировку $\|q\| = \|W^* w\| = \|w\| = 1$. Таким образом, максимум меры негауссовость для ξ достигается на векторе q, который имеет только один ненулевой (а следовательно единичный) элемент. Отсюда получаем

$$\arg\max_w \mathfrak{M}(w^* z) = W \arg\max_q \mathfrak{M}(q^* x) = W e_i = w_i, \qquad (3)$$

то есть весовой вектор w, доставляющий максимум меры негауссовости линейной комбинации главных компонент, является одним из столбцов матрицы W.

Итак, *ключевой идеей* метода является поиск такой линейной комбинации главных компонент $z(t)$, для которой степень негауссовости суммы максимальна. Коэффициенты этой суммы определяют столбец смешивающей матрицы W.

В качестве меры негаусовости рассматривают функции вида

$$\mathfrak{M}(\xi) = \mathfrak{M}(w^* z) = E[G(|w^* z|^2)],$$

где функция $G(\cdot)$ может подбираться исходя из особенностей конкретной задачи. Если $G(\xi) = \xi^2$, то мерой негауссовости является коэффициент эксцесса (kurtosis). Так называют статистику четвертого порядка, заданную формулой

$$\mathrm{kurt}(y) = E[|y|^4] - E[yy^*]E[yy^*] - E[yy]E[y^*y^*] - E[yy^*]E[y^*y],$$

что вместе с условием независимости действительной и мнимой частей $y \in \mathbb{C}$ означает

$$\mathrm{kurt}(y) = E[|y|^4] - 2\left(E[|y|^2\right)^2 - \left|E[y^2]\right|^2 = E[|y|^4] - 2.$$

Куртозис обращается в ноль для величин, имеющих нормальное распределение.

Эмпирически найдены и другие, часто более эффективные, выражения для функции G (ее называют функцей контраста или *контрастной функцией*). Когда мера негауссовости выбрана, строят алгоритм ее максимизации при ограничениях на w.

2.3. Максимизация контрастной функции. Максимизируем

$$\mathfrak{M}(w^*z) = \mathrm{E}[\mathrm{G}(|w^*z|^2)] \qquad \text{при} \quad \mathrm{E}[|w^*z|^2] = \|w\|^2 = 1.$$

Экстремум достигается в точке, где

$$\nabla \mathrm{E}[\mathrm{G}(|w^*z|^2)] - \beta \nabla \mathrm{E}[|w^*z|^2] = 0.$$

Градиент вычисляется отдельно по действительным и мнимым частям w.

$$\nabla \mathrm{E}[\mathrm{G}(|\xi|^2)] = \begin{pmatrix} \frac{\partial}{\partial w_{1r}} \\ \frac{\partial}{\partial w_{1i}} \\ \vdots \\ \frac{\partial}{\partial w_{nr}} \\ \frac{\partial}{\partial w_{ni}} \end{pmatrix} \mathrm{E}[\mathrm{G}(|w^*z|^2)] = 2 \begin{pmatrix} \mathrm{E}[\Re(z_1\xi^*)g(|\xi|^2)] \\ \mathrm{E}[\Im(z_1\xi^*)g(|\xi|^2)] \\ \vdots \\ \mathrm{E}[\Re(z_n\xi^*)g(|\xi|^2)] \\ \mathrm{E}[\Im(z_n\xi^*)g(|\xi|^2)] \end{pmatrix},$$

где, напомним, $\xi = w^*z$. Второе слагаемое раскрывается как

$$\nabla \mathrm{E}[|w^*z|^2] = 2 \begin{pmatrix} \Re(w_1) \\ \Im(w_1) \\ \vdots \\ \Re(w_n) \\ \Im(w_n) \end{pmatrix}$$

в силу ортогональности главных компонент $\mathrm{E}[zz^*] = \mathrm{I}$.

Для поиска точки экстремума мы применяем метод Ньютона. Матрица Якоби для $\nabla \mathrm{E}\left[\mathrm{G}(|w^*z|^2)\right]$ приближенно равна

$$\begin{aligned} \nabla^2 \mathrm{E}\left[\mathrm{G}(|w^*z|^2)\right] &= 2\mathrm{E}\left[(\nabla^2|w^*z|^2)g(|w^*z|^2) + \right. \\ &\qquad \left. + 2(\nabla|w^*z|^2)(\nabla|w^*z|^2)^\top g'(|w^*z|^2)\right] \approx \\ &\approx 2\mathrm{E}\left[g(|w^*z|^2) + |w^*z|^2 g'(|w^*z|^2)\right] \mathrm{I}, \end{aligned}$$

где $g = \mathrm{G}'$, а приближенное равенство получено разделением средних в предположении, что случайная величина и ее производная не коррелируют. Кроме того, мы воспользовались равенством $\mathrm{E}[zz^\top] = 0$, которое следует из некоррелированности действительной и мнимой части исходных источников. Матрица Якоби для второго слагаемого равна

$$\beta \nabla^2 \mathrm{E}[|w^*z|^2] = 2\beta \mathrm{I}.$$

Аппроксимация всего якобиана имеет вид

$$J = 2(E\left[g(|w^*z|^2) + |w^*z|^2 g'(|w^*z|^2)\right] - \beta)I.$$

Отметим, что якобиан диагонален и аналитически обратим. Таким образом, выращение для одной итерации Ньютона

$$w^+ = w - \frac{E\left[z(w^*z)^* g(|w^*z|^2)\right] - \beta w}{E\left[g(|w^*z|^2) + |w^*z|^2 g'(|w^*z|^2)\right] - \beta}$$

$$w_{\text{new}} = w^+ / \|w^+\|.$$

Домножая обе части получившегося уравнения на знаменатель, мы получаем более простое выражение

$$w^+ = E[z(w^*z)^* g(|w^*z|^2)] - E[g(|w^*z|^2) + |w^*z|^2 g'(|w^*z|^2)]w;$$

$$w_{\text{new}} = w^+ / \|w^+\|.$$

2.4. Алгоритм FastICA**.**

(0) Возьмем случайный вектор $w_{\{0\}}$ единичной нормы. Положим $k = 0$.

(1) Вычислим

$$w_{\text{new}} := E[z(w^*z)^* g(|w^*z|^2)] - E[g(|w^*z|^2) + |w^*z|^2 g'(|w^*z|^2)]w;$$

Отнормируем w_{new} на единицу.

(3) Если w_{new} достаточно близок к w

завершим алгоритм;

иначе

положим $w = w_{\text{new}}$ и вернемся на шаг 1.

В качестве меры близости весовых векторов используем угол между ними.

По завершении алгоритма w дает нам один из столбцов ортогональной смешивающей матрицы W. тем самым мы определяем одну независимую компоненту $x(t)$ как скалярное произведение

$$x(t) = w^* z(t).$$

Чтобы найти другие столбцы смешивающей матрицы и соответствующие независимые компоненты, воспользуемся унитарностью W. Новые столбцы лежат в ортогональном дополнении к подпространству найденных, поэтому достаточно внедрить в алгоритм шаг *ортогонализации* w_{new} к пространству уже найденных столбцов W.

(2) Ортогонализуем найденный вектор к пространству, заданному столбцами W

$$w_{\text{new}} := (I - WW^*)\, w_{\text{new}}.$$

Отнормируем w_{new} на единицу.

После завершения алгоритма W есть смешивающая матрица, а независимые компоненты определяются как $x(t) = W^* z(t)$, где $z(t)$ — главные компоненты.

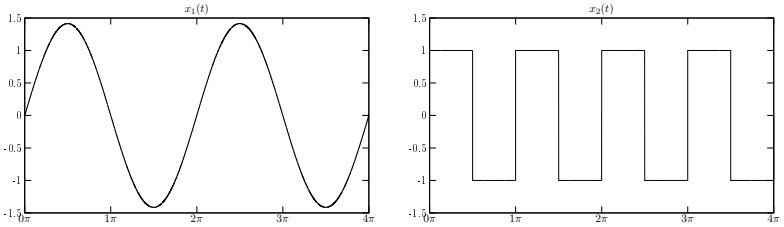
2.5. Пример. В заключении главы приведем классический пример задачи, с которой $\mathcal{ICA}$ справляется лучше, чем $\mathcal{PCA}$. Рассмотрим смеси двух независимых сигналов: синусоиды и прямоугольного меандра и попытаемся их разделить. Результаты эксперимента представлены на рис. 2 — без дополнительных комментариев ясно, что главные компоненты в принципе не восстанавливают источники, тогда как независимые компоненты восстанавливают их с точностью до масштаба и порядка.

3. Одновременная диагонализация многих статистик

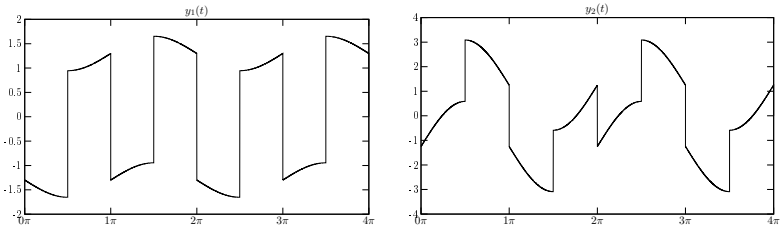
Изложение этой главы следует логике [25].

Рис. 2. Классический пример разделения синусоиды и меандра

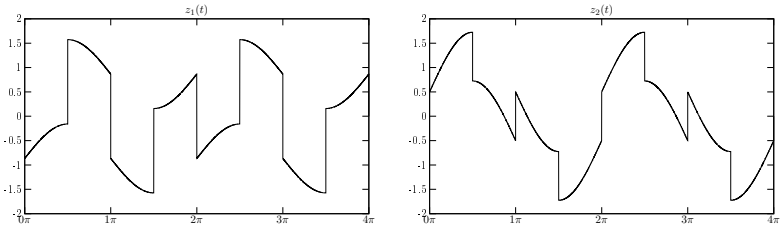
Источники



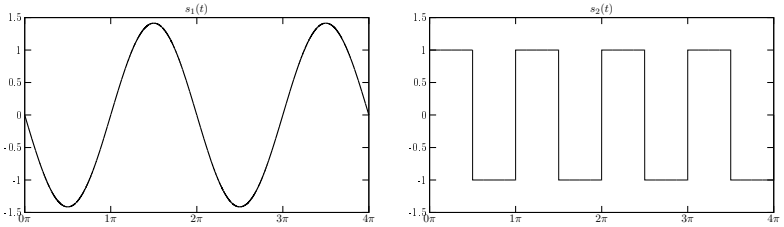
Сигналы (смеси источников)



Главные компоненты



Независимые компоненты FastICA



3.1. Почему две статистики лучше, чем одна, а больше двух — еще лучше. Использования только корреляционной статистики недостаточно, поскольку задача диагонализации одной матрицы имеет не единственное решение. Однако задача одновременной диагонализации двух матриц имеет в общем случае имеет единственное решение и, что также очень важно, разработаны стандартные алгоритмы решения этой задачи — обобщенной задачи на собственные значения. Возникает идея — использовать дополнительно к корреляционной статистике Φ_y *другую* статистику Q_y, диагонализуемую тем же преобразованием. Тогда задача определения смешивающей матрицы A из системы уравнений

$$\Phi_y = AD_\Phi A^*, \qquad Q_y = AD_Q A^*$$

имеет единственное решение.

Казалось бы, проблема закрыта. Тем не менее, выбрать две *различные* статистики оказывается не таким уж простым делом. Если Φ и Q «похожи», то точность решения обобщенной задачи на собственные значения значительно ухудшается — новая статистика на деле не приносит новой информации. Но как выбрать кросс-статистику так, чтобы она оказалась независима от первой? Априорного ответа, подходящего для всех случаев, по всей видимости, нет. Возможный в большинстве случаев ответ — взять большое число таких статистик и среди них обязательно окажется пара независимых.

В этом случае основным вычислительным ядром метода станет задача одновременного приведения к диагональному виду всех имеющихся кросс-статистик. При числе матриц более двух эта задача, вообще говоря, неразрешима. Однако в нашем случае существование решения гарантируется моделью $(*)$. Рассмотрим задачу об одновременной диагонализации p матриц (супер-обобщенную задачу на СЗ)

$$A_k = U\Lambda_k V^\top,$$

Перепишем ее в смысле минимизации нормы отклонения

$$\|A_k - U\Lambda_k V^\top\| \to \min$$

по всем неизвестным матрицам U, V и Λ_k.

Записав эту задачу в поэлементном виде

$$a_{ij}^{k} = \sum_{\alpha} u_{i\alpha}\lambda_{k\alpha}v_{j\alpha},$$

видим, что она равносильна задаче представления трехмерного массива $\mathcal{A} = [a_{ijk}]$ в виде

$$a_{ijk} = \sum_{\alpha} u_{ia}v_{ja}w_{ka},$$

где столбцы матрицы W содержат элементы диагональных матриц Λ_k. Возникла задача *трилинейного разложения* трехмерного массива данных (тензора). Возможные подходы к ее решению описаны в [23, 24, 25] и [29]-[36].

3.2. Какие кросс-статистики можно использовать. Рекомендуемый метод выбора Q зависит от характеристик рассматриваемых сигналов.

3.2.1. Нестационарные сигналы. Если сигнал нестационарен, то его осреднение по различным отрезкам (окнам) времени приводит к различным статистикам

$$Q_y(t) = E_{(t:t+\Delta t)}[y(t)y^*(t)] = AE_{(t:t+\Delta t)}[x(t)x^*(t)]A^* = AQ_x(t)A^*.$$

Выбрав $Q_y = Q_y(t)$ для какого-то конкретного t, или взяв в качестве Q_y случайную линейную комбинацию $Q_y(t)$ для различных моментов t, мы получаем новую статистику. Вообще говоря, если она оказывается близка к исходной статистике Φ_y, то ее использование не дает нам новой информации, и задача поиска обобщенного собственного разложения становится плохо определенной. В этом случае можно использовать набор матриц $Q_y^k = Q_y(t_k)$ для цепочки моментов t_k и свести задачу к построению одновременной диагонализации для *всех* матриц Q_y^k.

3.2.2. Цветные сигналы. Если сигнал не является белым, то возможно использование следующей кросс-корреляцией

$$Q_y(\tau) = E[y(t)y^*(t+\tau)] = AE[y(t)y^*(t+\tau)]A^* = AQ_x(\tau)A^*.$$

Если эта автокорреляция для источника не равна нулю, то эта статистика предоставляет новую информацию при выборе одного или нескольких различных сдвигов τ.

3.2.3. Не-Гауссовы сигналы. Если сигнал является стационарным и не обладает автокорреляцией, то статистики для различных t и τ не дают новой информации. В этом случае можно воспользоваться статистикой четвертого порядка, называемой *кумулянтом*

$$(\mathcal{C}_y)_{ijkl} = \mathrm{E}[y_i\bar{y}_j y_k\bar{y}_l] - \mathrm{E}[y_i\bar{y}_j]\mathrm{E}[y_k\bar{y}_l] - \mathrm{E}[y_i y_k]\mathrm{E}[\bar{y}_j\bar{y}_l] - \mathrm{E}[y_i\bar{y}_l]\mathrm{E}[\bar{y}_j y_k].$$

Кумулянт суммы независимых величин равен сумме кумулянтов. Для сигнала, имеющего нормальное распределение, кумулянт равен нулю. Поэтому при применении статистики четвертого порядка происходит автоматическая фильтрация шумов, возникающих в принимающих устройствах, или случайных погрешностей измерения, если они имеют нормальное распределение.

Статистики второго и четвертого порядка Φ_y и $\mathcal{C}_y$, отвечающие наблюдаемым сигналам $y(t)$, связаны со статистиками источников следующими соотношениями[def)]

$$\Phi_y = A\Phi_x A^*, \qquad \mathcal{C}_y = \mathcal{C}_x \times_1 A \times_2 \bar{A} \times_3 A \times_4 \bar{A}. \tag{4}$$

Из независимости источников следует диагональность статистик Φ_x и $\mathcal{C}_x$, причем второе условие означает, что тензор имеет ненулевые элементы только при $i = j = k = l$, и, очевидно, предоставляет существенно больше информации для определения A. Более того, в общем случае второе уравнение системы (4) не имеет точного решения и рассматривается в смысле минимизации нормы отклонения целевого тензора от ближайшего диагонального.

$$A = \arg\min_{B,\mathcal{D}} \left\| \mathcal{C}_y - \mathcal{D} \times_1 B \times_2 \bar{B} \times_3 B \times_4 \bar{B} \right\| \tag{diag}$$

Решив задачу в этой формулировке, можно даже получить *сверхразрешение,* то есть выделить из n наблюдаемых смесей m исход-

[def)] Тут символом «$\times_p$» обозначено умножение тензора на матрицу вдоль p–го индекса. Например для тензора $\mathcal{A} = [a_{ijkl}]$ размера $n \times n \times n \times n$ и матрицы B размера $m \times n$ результатом операции «$\times_2$» является тензор $\mathcal{C} = [c_{ijkl}]$ размера $n \times m \times n \times n$, поиндексно определенный равенством

$$c_{ijkl} = \sum_{j'=1}^{n} a_{ij'kl} b_{jj'}.$$

ных независимых компонент при $m > n$. Однако алгоритм решения оказывается в несколько раз сложнее, чем метод одновременной диагонализации матриц. Есть возможность пожертвовать сверхразрешением и свести задачу к более простой. Для этого рассмотрим тензор $\mathcal{C}_z$ размера $m \times m \times m \times m$, как семейство из m^2 матриц, объединив последние два индекса в один

$$\tilde{\mathcal{C}} = [(c_{ij})_\alpha] = (\mathcal{C}_z)_{ij(kl)}, \qquad (kl) \leftrightarrow \alpha.$$

Заменим задачу суперсимметричной диагонализации четырехмерного тензора $\mathcal{C}$ более простой задачей одновременной диагонализации семейства m^2 матриц C_α. На языке тензорных операций это означает, что вместо задачи $(diag)$ мы рассматриваем

$$A = \arg \min_{B,Q,\mathcal{D}} \left\| \tilde{\mathcal{C}} - \mathcal{D} \times_1 B \times_2 \bar{B} \times_3 Q \right\| \qquad (diag')$$

очевидно, накладывающую ослабленные требованния на A, поскольку мы пренебрегаем структурой матрицы Q и «забываем» о ее связи с B. Новая задача имеет единственное решение, однако только при числе источников не больше числа приемников.

4. Реализация алгоритмов $\mathcal{BSS}$ в пакетном режиме

4.1. Требования к пакетной версии алгоритмов. Под работой алгоритмов в пакетном режиме мы понимаем следущее. Пусть общая протяженность принятых сигналов T_{full} но в вычислительную систему для обработки они поступают N пакетами длины T (для простоты изложения будем полагать, что длины всех пакетов одинаковы — даже если это не так, суть метода не меняется). Эта модель работы с данными может потребоваться по одной из следующих причин.

- Нет возможности ждать достаточно долго, пока весь сигнал длины T_{full} будет принят. Напротив, требуется получить предварительные данные о сигналах как можно быстрее (пусть и со значительной погрешностью, вызванной малой длиной накопленной информации), и затем только уточнять их по новым фрагментам полученных сигналов.

- Оперативной памяти вычислительного устройства и/или его вычислительной мощности может быть недостаточно для запуска алготитма разделения сигналов длины T_{full}. При этом требуется принудительно расщепить рассматриваемый сигнал на несколько меньших частей, которые могут быть обработаны на имеющемся вычислительном устройстве, и работать с ними последовательно.

- В условиях реальных вычислений положение источников и/или приемников и/или состояние среды, в которой распространяется сигнал, могут меняться. Даже если эти изменения достаточно медленные, они оказывают влияние на смешивающую матрицу и точность выполнения основной линейной модели $(*)$. В этом случае нет смысла использовать всю информацию о сигнале длины T_{full}, так как нарушение линейной модели вносит неустранимую погрешность в результат работы алгоритма. Необходимо оперировать отрезками сигнала длины T, такими что на протяжении этого времени состояние системы и следовательно элементы смешивающей матрицы A могут с удовлетворительной точностью считаться постоянными.

При реализации алгоритмов слепого разделения сигналов в пакетном режиме (будем называть их пакетными версиями алгоритмов или кратко *пакетными алгоритмами*) мы хотели бы по возможности добиться выполнения следующих условий.

- **Накопление информации.** После того, как пакетный алгоритм получил N пакетов данных длины T, точность разделения сигналов должна быть такой же, как и у исходного алгоритма, обработавшего сингал длиной $T_{full} = NT$.

- **Сохранение устойчивости.** Если при обработке какого-то из полученных пакетов пакетный алгоритм не достигает сходимости (или не завершается корректно по любой иной причине), это не должно останавливать алгоритм в целом. Текущий пакет данных должен быть разделен с использованием размешивающей матрицы W с прошлого шага, и алгоритм должен быть продолжен.

- **Связность результата**. Поскольку источники определяются с точностью до множителя и порядка, алгоритм в пакетной версии должен позаботиться о том, чтобы разделенные сигналы в двух соседних пакетах были *сшиты,* то есть на границе раздела пакетов не наблюдалось скачка амплитуды, фазы источников или перестановки выдаваемых фрагментов источников местами.

В следующих разделах мы покажем, для каких методов можно добиться выполнения указанных требований.

4.2. Почему алгоритм FastICA не накапливает информацию. Алгоритм FastICA (и родственные ему) вычисляет размешивающую матрицу W, решая экстремальную задачу относительно меры $\mathfrak{M}(w^*z(t))$. Предположим, что передача ведется двумя пакетами по T элементов в каждом. Получив первый пакет $y^{[1]}(t)$, мы переходим в базис главных компонент $z^{[1]}(t)$ и применяя алгоритм FastICA, находим текущее приближение к смешивающей матрице $W^{(1)}$. Однако после получения второго пакета $y^{[2]}$ (и что важно, потеряв вектора $y^{[1]}(t)$ и $z^{[1]}(t)$), мы не можем максимизировать

$$\mathfrak{M}(w^*z(t)) = \mathfrak{M}(w^*z^{[1]}(t)) + \mathfrak{M}(w^*z^{[2]}(t)),$$

поскольку не можем вычислить первое слагаемое для произвольного w. Если отдельно максимизировать только второе слагаемое, точность определения матрицы $W^{(2)}$ на втором этапе пакетного алгоритма окажется не выше, чем на первом шаге — поступившая новая информация не позволяет уточнить элементы смешивающей матрицы. В этом смысле мы говорим, что не существует пакетной версии алгоритма FastICA, накапливающей информацию в ходе работы.

Тем не менее, метод FastICA может быть полезен, особенно в случае, когда используемый размер пакетов достаточно велик, а стабильность матрицы A при переходе от пакета к пакету вызывает сомнения.

В следующих разделах мы покажем, что методы, основанные на диагонализации различных статистик принятых сигналов, обладают приятным свойством накопления информации, и построим соответствующие пакетные алгоритмы.

4.3. Пакетная версия метода главных компонент. В отличие от методов типа Fast$\mathcal{ICA}$, алгоритм $\mathcal{PCA}$ допускает пакетную реализацию. Принципиальным является тот факт, что статистики, в отличие от меры $\mathfrak{M}(w^*z(t))$, не зависят от определяемых величин и могут аддитивно обновляться в ходе работы пакетного метода. Для вычисления статистики на «большом» отрезке времени (например на полной длине приема T_{full}) нам достаточно просуммировать с некоторыми коэффициентами статистики с «малых» фрагментов времени (например статистики по пакетам длины T.) При этом коэффициенты зависят, естественно, от длин соответствующих статистик и от номера принятого пакета. Например, если уже насчитана статистика

$$\Phi_y^{(k)} := E[y(t)y^*(t)]_{[0:kT]},$$

то статистика $\Phi_y^{(k+1)}$ на следующем шаге пакетного алгоритма определится как

$$\begin{aligned}\Phi_y^{(k+1)} &:= \tfrac{k}{k+1}\Phi_y^{(k)} + \tfrac{1}{k+1}E[y(t)y^*(t)]_{[kT:kT+T]} = \\ &= \tfrac{k}{k+1}\Phi_y^{(k)} + \tfrac{1}{k+1}E[y^{[k+1]}(t)y^{[k+1]}(t)^*].\end{aligned}$$

4.4. Обновление статистик для сигналов. Простой по существу метод обновления статистики, предложенный для $\mathcal{PCA}$, можно обобщить, сформулировав процедуру следуюшим образом.

Алгоритм 1. *(Обновление статистики $\Psi_y = E[G(y)]$).*

1. Получить $y^{[1]}$. Вычислить $\Psi_y^{(1)}$.

...

k. Получить $y^{[k]}$. Вычислить статистику $E[G(y^{[k]})]$ на текущем отрезке времени. Вычислить $\Psi_y^{(k)}$ по формуле

$$\Psi_y^{(k)} = \alpha_k\Psi_y^{(k-1)} + (1-\alpha_k)E[G(y^{[k]})]. \qquad (5)$$

В зависимости от схемы выбора α_k алгоритм обновления может трактоваться различным образом.

- $\alpha_k = 0$ — модель абсолютно неустойчивой среды. Данные с прошлых шагов не используются, разделение ведется по каждому новому пакету независимо от информации о предыдущих.

- $\alpha_k = \frac{k-1}{k}$ — модель устойчивой среды, уточнение размешивающей матрицы на каждом шаге. Происходит точное обновление статистик, полученная $\Psi^{(k)}$ совпадает с усреднением за период $[0 : kT]$. За счет этого на каждом шаге уменьшается статистическая погрешность, вносимая конечным размером выборки, и повышается точность вычислений смешивающей матрицы — конечно, если искомая смешивающая не меняется на протяжении k полученных пакетов.

- $\alpha_k = 1$ — модель устойчивой среды, быстрое размешивание. Новая иформация не используется, статистики (а вместе с ними и размешивающая матрица) копируются с первого этапа. Предполагается, что смешивающая матрица не меняется, и размера статистики на первой выборке длины T достаточно для восстановления размешивающей матрицы с необходимой точностью. Достоинство этой модели в очень быстром выполнении алгоритма на втором и последующих этапах — по сути, происходит только лишь умножение нового пакета сигналов на размешивающую матрицу, взятую с первого этапа.

Все три схемы выбора α_k хороши в рамках своей модели и имеют свои достоинства. Безусловно, возможны и другие схемы выбора параметра α_k, отвечающие иным предположениям об скорости изменения элементов смешивающей матрицы. Как видно из рассмотреных примеров, выбор α_k оказывает огромное влияние на работу и результаты алгоритма; тем не менее, мы не можем предложить какой-то одной, «действительно наилучшей» стратегии выбора α_k. Мы полагаем, что этот вопрос может решаться следующими способами:

- аналитически — при известной степени неустойчивости смешивающей матрицы, когда хорошо известны характеристики принимаемых сигналов;

- эмпирически — при работе в режиме «лаборатории», когда мы имеем дело с записанными сигналами, не знаем степень устойчивости среды во время их получения, но можем провести серию воспроизводимых экспериментов с разными параметрами α_k, имея на них проведение достаточно большое время;
- решением оператора — когда характеристики устойчивости среды и принимаемых сигналов неясны, а принимаемый сигнал должен обрабатываться практически в режиме реального времени. В этом случае имеет смысл придерживаться стратегии накопления информации $\alpha_k = (k-1)/k$ столь долго, пока качество разделения источников выглядит удовлетворительным и не ухудшается с каждым новым пакетом данных. Если качество разделения ухудшается (разделенные источники хуже декодируются или прослушиваются), следует «очистить историю», установив $\alpha_k = 0$ и начать накопление данных заново.

4.5. Алгоритмы в пакетном режиме. Все вышесказанное подготовило почву для того, чтобы изложить пакетные версии *всех* алгоритмов, использующих статистики сигналов, в единообразном виде.

Алгоритм 2. *(Прототип пакетного алгоритма)*

Вход: пакеты сигналов $y^{[k]}$, $k = 1, \ldots, N$.

Выход: пакеты источников $x^{[k]}$ и смешивающая матрица.

Параметры: статистики $\Psi_y = E[G_p(y)]$, $p = 1, \ldots, n_p$, схема выбора параметров α_k.

[Описан k–тый шаг алгоритма]

- Получить $y^{[k]}$.
- Вычислить статистики $E[G_p(y^{[k]})]$ на текущем отрезке времени.
- Обновить их по формуле (5) с помощью выбранных α_k.
- Запустить алгоритм слепого разделения сигналов по статистике $\Psi_y^{(k)}$. Найти размешивающую матрицу $W^{(k-1)}$

- Разделить источники $x^{[k]} = W^{(k)} y_{(k)}$.
- «Склеить» источники $x^{[k]}$ с предыдущими пакетами $x^{[k-1]}$.

Последний этап нами пока не обсуждался, а между тем именно он обеспечивает выполнение третьего требования к пакетной версии алгоритмов — связности результата. Посвятим ему отдельный раздел.

4.6. Поддержание связности. «Разрыв» может возникать по двум причинам:

- после получения нового пакета данных активная смешивающая матрица изменилась так, что источники поменялись местами;
- после получения нового пакета данных амплитуда и фаза источников поменялась так, что наблюдается скачок на границе раздела пакетов.

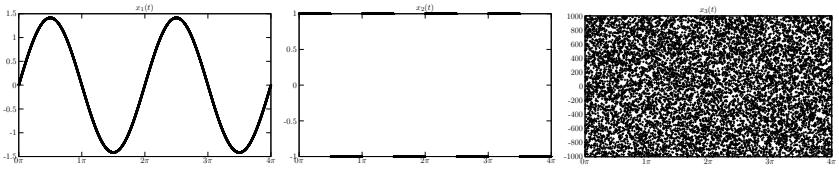
Для решения первой проблемы — перестановки сигналов — можно предложить два способа. Первый состоит в том, что рассматриваются две матрицы: размешивающая с прошлого шага и смешивающая с текущего. Их произведение

$$W^{[k-1]} A^{[k]} = P_k$$

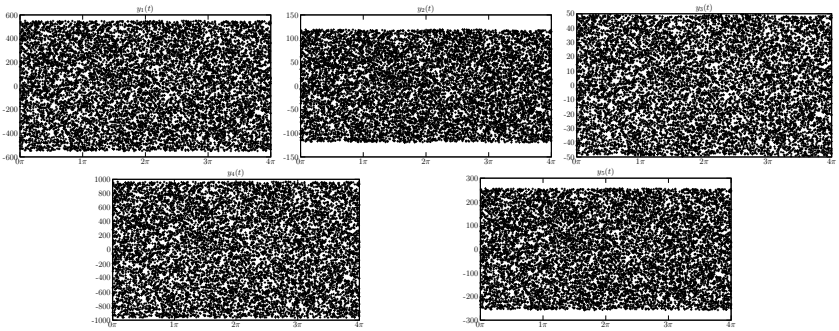
должно быть близко к масштабированной матрице перестановки, если только смешивающая матрица не меняется слишком сильно на отрезке времени порядка размера пакета. Эта матрица и указывает, как нужно упорядочить новые пакеты источников по отношению к имеющимся. Есть и другой подход, возможно, более точный, который состоит в том, чтобы рассматривать пакеты данных с некоторым небольшим наложением. Сохранив это небольшое количество элементов из прошлого пакета источников, мы можем подобрать коэффициенты масштабирования так, чтобы получить наилучшее совпадение фрагмета источников из прошлого и текущего пакетов в их области наложения. Однако этот метод в известном смысле усложняет логику программы.

Рис. 3. Два сигнала и сильный шумовой источник

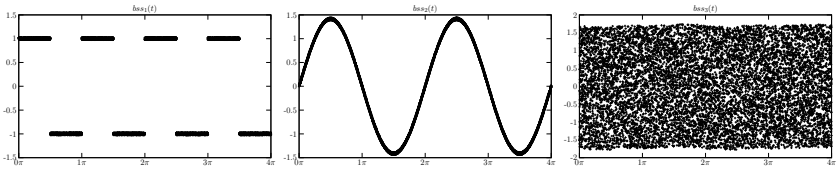
Смешиваем синусоидальный сигнал единичной нормы, прямоугольный меандр и шум порядка 10^3. *Размер выборки большой* $l = 10^4$ *отсчетов*



Компоненты смеси выглядят, как шум



Тем не менее, метод BSS позволяет выделить источники из смеси



На выборке $l = 100$ *точность вычислений ниже*

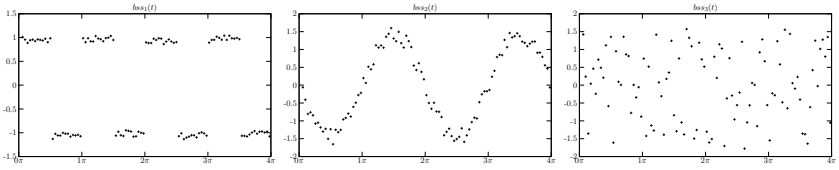
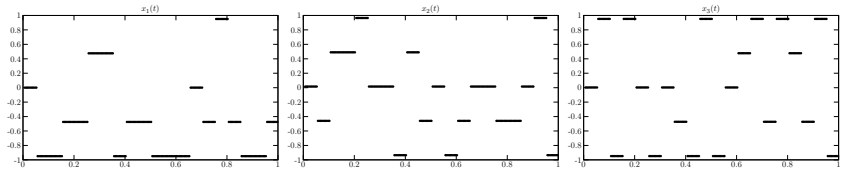
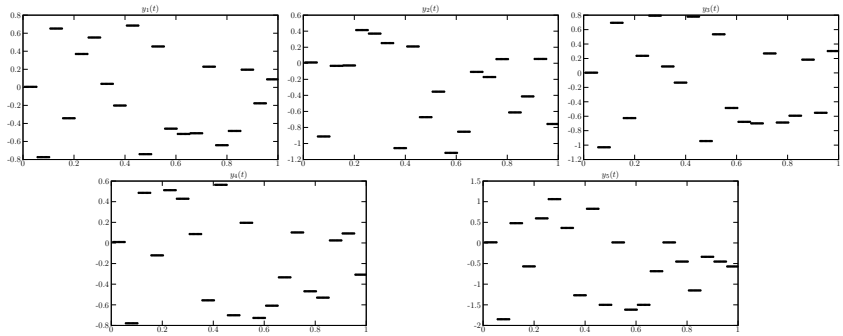


Рис. 4. Три сигнала с модуляцией типа qpsk

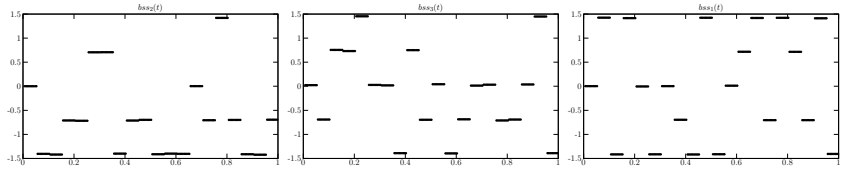
Смешиваем три дискретных четырехпозиционных сигнала. Размер выборки большой $\mathrm{l} = 10^4$ *отсчетов*



Компоненты смеси выглядят достаточно сложно



Тем не менее, разделение сигналов происходит удачно



5. Примеры

Рассмотрим несколько простых примеров разделения сигналов для иллюстрации возможностей метода $\mathcal{BSS}$. На рисунке 3 показан эксперимент по выделению сигналов из-под сильного шумового источника. В качестве сигналов взяты традиционные синусоида и прямоугольный меандр. Обратите внимание, что при разделении сигналов на малой выборке точность определения компонент значительно снижается.

На рисунке 4 показано разделение нескольких сигналов в модуляцией типа QPSK. Модуляция QPSK кодирует информацию, устанавливая фазу передаваемого сигнала в одну из четырех позиций. На рисунках — аналог этих сигналов в действительной арифметике, более удобный для графического отображения. Видим, что задача о разделении смеси трех кодированных сигналов может быть с успехом решена описанными методами $\mathcal{BSS}$. На практике она возникает при передачи информации в MIMO каналах, например, в новых стандартах сотовой связи.

Список литературы

[1] Cherry E. C. Some experiments on the recognition of speech, with one and with two ears // *Journal of Acoustical Society of America.* 1953. V 25(5). P. 975–979.

[2] Hämäläinen M., Hari R., Ilmoniemi R., Knuutila J. and Lounasmaa O.V. Magnetoencephalography — theory, instrumentation, and applications to noninvasive studies of signal processing in the human brain // *Reviews of Modern Physics.* 1993. V. 65. P. 413–497.

[3] De Lathauwer L., De Moor B., Vandewalle J. Fetal electrocardiogram extraction by source subspace separation ss *Proc. of IEEE Signal Processing / Athos Workshop on Higher-Order Statistics,* Girona, Spain. 1995. P. 134-138.

[4] Vigário R. N. Extraction of ocular artifacts from EEG using independent component analysis // *Electroenceph. clin. Neurophysiol.* 1997. V. 103. P. 395–404.

[5] Vigário R., Särelä J., and Oja E. Independent component analysis in wave decomposition of auditory evoked fields // *Proc. Int. Conf. on Artificial Neural Networks (ICANN'98)* 1998. P. 287–292.

[6] Lei Xie, Jun Wu. Global optimal ICA and its application in MEG data analysis. // *Neurocomputing.* 2006. V. 69. P. 2438–2442.

[7] B. Chen and A. P. Petropulu. Frequency domain MIMO system identification based on second and higher-order statistics // *IEEE Trans. Signal Processing.* 2001. V. 49, №8. P. 1677–1688.

[8] De Lathauwer L., Comon P., De Moor B., Vandewalle J. ICA algorithms for 3 sources and 2 sensors // *Proc. of the IEEE Signal Processing Workshop on Higher-Order Statistics (HOS'99),* Caesarea, Israel. 1999. P. 116–120.

[9] De Lathauwer L., De Moor B., Vandewalle J. An algebraic approach to blind MIMO identification // *Proc. of the 2nd Int. Workshop on independent component analysis and blind source separation (ICA 2000),* Helsinki, Finland. 2000. P. 211–214.

[10] R. Bro. Review on Multiway Analysis in Chemistry — 2000–2005. // *Analytical Chemistry.* 2006. V. 36. P. 279.

[11] Back A. D., Weigend, A.S. What drives stock returns? — An independent component analysis // *Proc. of the IEEE/IAFE/INFORMS Conf. on Comp. Intelligence for Financial Engineering (CIFEr).* 1998. P. 141–156.

[12] Moody J., Howard, Y. Term structure of interactions of foreign exchange rates // *Computational Finance — Proc. of the* 6^{th} *Int'l Conf.* 1999. P. 24–35.

[13] Draper B. A., Baek K., Bartlett M. S., Beveridge J. R. Recognizing faces with PCA and ICA // *CVIU(91),* 2003. №1-2. P. 115-137.

[14] Jongsun Kim, Jongmoo Choi, Yuneho Yi. Biometric Authentication — Springer Berlin, 2004.

[15] Hotelling H. Analysis of a Complex of Statistical Variables with Principal Components // *Journal of Educational Psychology.* 1933. V. 24. P. 417–441.

[16] Tetsuo Sato. Application of principal-component analysis on near-infrared spectroscopic data of vegetable oils for their classification // *Journal of the American Oil Chemists' Society.* 1994. V. 71, №3. P. 293–298.

[17] Aiyi Liu, Ying Zhang, Gehan E., Clarke R. Block principal component analysis with application to gene microarray data classification // *Statistics in Medicine.* 2002. V. 21. P. 3465–3474.

[18] Hyvärinen A. Fast and robust fixed-point algorithm for independent component analysis // *IEEE Trans. on Neural Network.* 1999. V. 10, №3. P. 626-634.

[19] E. Bingham and A. Hyvärinen. A fast fixed-point algorithm for independent component analysis of complex-valued signals. // *Int. J. of Neural Systems.* 2000. V. 10, №1. P. 1–8.

[20] Comon P. Independent Component Analysis, a new concept? // *Signal Processing, Elsevier.* 1994. V. 36, №3. P. 287–314.

[21] Hyvärinen A., Erkki Oja. Independent Component Analysis: Algorithms and Applications // *NEural Networks.* 2000. V. 13, №4-5. P. 411-430.

[22] St ogbauer H., Kraskov A., Astakhov S. A., Grassberger P. Least Dependent Component Analysis Based on Mutual Information // *Phys. Rev.* 2004. E 70, 066123.

[23] Cardoso J.-F., Souloumatic A. Blind beamforming for non Gaussian signals // *IEE Proc.-F.* 1993. V. 140, №6. P. 362–370.

[24] Cardoso J.-F., Souloumatic A. Jacobi angles for simultaneous diagonalisation // *SIAM J. Mat. Anal. Appl.* 1996. V. 17 №1. P. 161–164.

[25] Parra L., Sajda P. Blind Sourse Separation via Generalized Eigenvalue Decomposition // *J. of Machine Learning Research.* 2003. V. 4. P. 1261–1269.

[26] De Lathauwer L., De Moor B., Vanderwalle J. Computation of the canonical decomposition by means of a simultaneous generalized Schur decomposition // *SIAM J. Matrix Anal. Appl.* 2004. V. 26. P. 295-227.

[27] Ziehe A., Laskov P., Müller K.-R., Nolte G. A linear least-squares algorithm for joint diagonalization // *Proc.* 4^{th} *Intern. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003).* 2003. P. 469–474.

[28] Ziehe A., Kawanabe M., Hamerling S., Müller K.-R. A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation // *Journal of Machine Learning Research.* 2004. V. 5. P. 801-818.

[29] Оселедец И.В., Савостьянов Д.В. Методы разложения тензора // *Матричные методы и технологии решения больших задач Сб. под ред. Е. Е. Тыртышникова* С. 51–64.

[30] Оселедец И.В., Савостьянов Д.В. Трехмерный аналог алгоритма крестовой аппроксимации и его эффективная реализация // *Матричные методы и технологии решения больших задач Сб. под ред. Е. Е. Тыртышникова* — ИВМ РАН, 2005. С. 65–99.

[31] Оселедец И.В., Савостьянов Д.В. Быстрый алгоритм для одновременного приведения матриц к треугольному виду и аппроксимации тензоров // *Матричные методы и технологии решения больших задач Сб. под ред. Е. Е. Тыртышникова* — ИВМ РАН, 2005. С. 101–116.

[32] Оселедец И.В., Савостьянов Д.В. Об одном алгоритме построения трилинейного разложения // *Матричные методы и технологии решения больших задач Сб. под ред. Е. Е. Тыртышникова* — ИВМ РАН, 2005. С. 117–130.

[33] Оселедец И.В., Савостьянов Д.В. Минимизационные методы аппроксимации тензоров и их сравнение // *ЖВМиМФ*. 2006. Т. 46, №10. С. 1641-1650.

[34] Oseledets I. V., Savostianov D. V., Tyrtyshnikov E. E. Tucker dimensionality reduction of three-dimensional arrays in linear time // *SIMAX,* принято к публикации.

[35] Oseledets I. V., Savostianov D. V., Tyrtyshnikov E. E. Fast simultaneous ortogonal reduction to triangular matrices. // *SIMAX,* принято к публикации.

[36] Савостьянов Д. В. Быстрая полилинейная аппроксимация матриц и интегральные уравнения Дисс. на степ. к.ф.-м.н. — ИВМ РАН, Москва, 2006.